PACIS 2016 Proceedings

Pacific Asia Conference on Information Systems (PACIS)

Summer 6-27-2016

IS PREVENTION ALWAYS BETTER? A CASE OF IT SERVICE MANAGEMENT

Prathamesh Mayekar
Tata Consultancy Services, prathamesh.mayekar@tcs.com

Veerendra K. Rai
Tata Consultancy Services, veerendrak.rai@tcs.com

Abhinay Puvvala
Tata Consultancy Services, abhinay.puvvala@tcs.com

Harrick M. Vin
Tata Consultancy Services, harrick.vin@tcs.com

IS PREVENTION ALWAYS BETTER? A CASE OF IT SERVICE MANAGEMENT

Prathamesh Mayekar, Digitate, Tata Consultancy Services, Pune, India, prathamesh.mayekar@tcs.com

Veerendra K. Rai, Digitate, Tata Consultancy Services, Pune, India, veerendrak.rai@tcs.com

Abhinay Puvvala, Digitate, Tata Consultancy Services, Pune, India, abhinay.puvvala@tcs.com

Harrick M.Vin, Digitate, Tata Consultancy Services, Pune, India, harrick.vin@tcs.com

Abstract

Information Technology Infrastructure Library (ITIL), a framework for IT Service Management (ITSM), emphasizes the need for an ongoing preventive activity woven into the fabric of enterprise IT of organizations as opposed to a reacting to a specific situation. However, with the increasing focus on cost reduction, it is essential to revisit the trade-off between costs and other primary ITSM objectives such as service availability and quality. With this basic premise, we compare the cost of conducting IT service operations with varying levels of prevention. We modelled the IT service operation processes based on queuing and software reliability theories while assessing the impact of exogenous variables such as initial application maturity, drop rates & monitoring cost. We illustrated that optimum lies between the extremes of complete prevention and reaction. Also, we were able to observe the pronounced impact of staffing stickiness on the results.

Keywords: IT Service Management, Monitoring & Control, Cost of Quality, Queuing Theory, Software Reliability.

1 INTRODUCTION

Imagine a human child immunized against all diseases and abnormalities once and for all. Unfathomable bliss of life lived with perfect health and uncountable amount of money saved. That is the power of prevention. Human societies across nations and geographies have extolled the virtue of prevention in all walks of life for ages. It is better to 'prevent' wherever possible than to maintain complex systems, structures and processes to deal with the aftermath (Papazoglou 2008).

Same applies to much smaller and mundane problem in the domain of IT production service support also referred to as IT operations management and IT service management. Everything cannot be prevented in reality, proactively. So, we need to 'react' to the situations which could not be prevented. There is a cost associated with preventing a situation from taking place as well as reacting to the situation once it has taken place. Given the limited resources available to managers, they have to decide between investing on monitoring and control system to prevent and incurring the additional costs for firefighting. Understanding this trade-off between prevention and reaction is the central theme of this paper.

An incident is essentially a service outage which can either be prevented or attended to. Prevention, like imposing a prior regulation standard or implementing an ex-ante policy, involves putting together a monitoring and control system which continuously monitors for deviations and generates alerts as and when needed. These alerts are handled and sent for possible root cause analysis to avoid recurrence. A reactive approach works to restore the service at the earliest possible, analysing the incident to determine root cause and implementing an appropriate fix to avoid repeat occurrences.

ITIL (Cannon et al. 2007) segregated the above mentioned activities in preventive and reactive approaches into clearly defined processes – Event management, Incident management, Problem management and Change management. Foreseeing application outages by the monitoring and control system to take preventive measures (Event management), handling outages to quickly restore service (Incident management), conducting root cause analysis to determine the errors behind outages/alerts (Problem management) and implementing suitable fixes/changes to fix the bugs (Change management) are the four processes that define the scope of the issue we are addressing in this paper.

The paper is organized as follows- Section 2 describes relevant literature on preventive & proactive management in other domains and the methodology used. Section 3 describes the mathematical model, various key components of the model and the closed form solution for determining optimal monitoring level. Section 4 has the results, which includes analysis on the movement of various costs with the level of prevention and sensitivity analysis of the optimal prevention level with various input parameters. Section 5 concludes the paper.

2 LITERATURE REVIEW

Similar studies have been conducted in various other domains. Economics literature has focussed on comparing the ex-ante policies that come into effect before the accident happened and ex-post policies that come into effect after the accident to control the occurrence of accidents. Shavell (1984) has examined the routes of safety regulation and liability to effectively control risks due to accidents. The study suggests that neither of the two extremes is effective in controlling the risks of accidents whereas their joint use is more beneficial. Kolstad et al. (1990) describes accidents as hazardous economic activities and proposes the right mix of policies to regulate them. The findings imply that the exclusive use of ex-ante or ex-post policies would be sub optimal. A similar study (Wittman 1977) to compare post liability versus prior regulation suggests the optimal policy choice is dependent on factors such as information availability, insurance and transaction costs.

Supply chain literature has off late seen a spurt in studies (Kleindorfer & Saad 2005, Hendricks & Singhal 2005, Chopra & Sodhi 2004) on disruption risk management, which focuses on designing the right policy framework for handling disruptions due to arising from natural hazards, terrorism, political instability, equipment malfunctions and other unforeseen discontinuities in supply. These studies address the balance between minimizing risks due to these disruptions while minimizing the overall costs.

Our study bases itself on the Juran's (1962) cost of quality (COQ) model. Juran's model concerns itself with the quality of products, services and processes. Broadly, the two extremes of quality are when the product is faultless, in which case the failure costs are zero and conformance costs incurred are very high and when the product is 100% defective, in which case failure costs rise to infinity and conformance costs incurred are zero. Although in a long run, Juran's COQ model prefers perfection (100% conformance), but for a finite time horizon it states that the optimal conformance level lies between these two extremes of quality. Analogous to an accident' or 'failure' in the context of IT Service Management is an 'incident'.

We have modelled the major IT service operation processes as a queuing network. All our analysis is done for the optimal staffing level based on this queuing network for varying workload. This methodology is somewhat similar to the one used by Kolesar et al. (2007) and Whitt et al. (1999).

3 PROBLEM APPROACH

We have modelled the major IT service operation processes (Incident management, Event management, Problem management and Change Management) as a queuing network. We look at service engagement over time horizon T. One unit of time corresponds to a staffing interval. After every unit service provider can look at arrival rates of incidents and events and update the resource requirement accordingly.

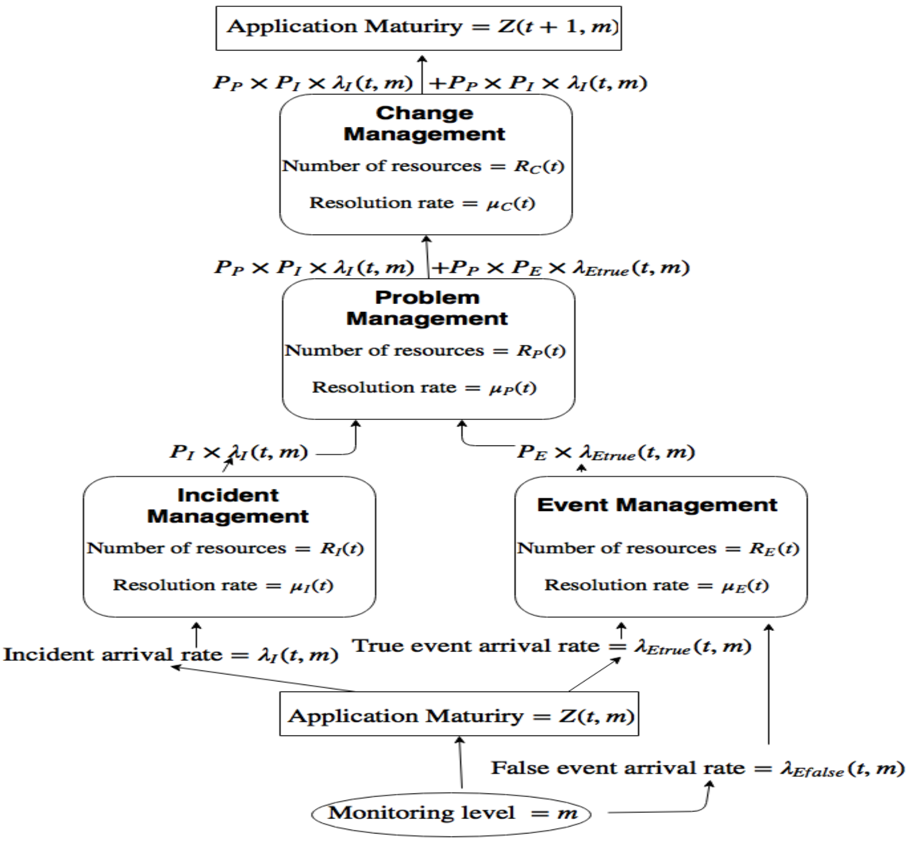

Figure 1. Problem Approach

Figure 1 explains our problem approach. Incidents and events arrive. An attenuated version of this incidents and events are passed onto problem management. Some of these problems are passed on for change management which eventually leads to changes. We assume that all the changes made in one staffing interval are updated at the end of staffing interval. Usually, patches that comprise of multiple changes are installed periodically. Hence, the incident and event arrival rates remain the same in a staffing interval.

3.1 Event Management

An event is any noticeable occurrence that has significance for IT infrastructure management and IT service delivery and service quality (Cannon et al. 2007). Events are typically notifications created by an IT service, a configurable item or monitoring tool. Monitoring is a vital cog in the Event management process. For example, monitoring an Enterprise IT system involves continuously polling the system at different levels for various key metrics such as disk utilization, memory utilization, processor utilization (Operating System level) database size, listener, active connections, running tasks, failed tasks (Database level) and memory size, http requests, services deployed (Application server level). Monitoring is modelled on an aggregate level as follows.

Monitoring Level m: The monitoring level m can be defined as the amount of prevention done for a particular service. We use monitoring level and level of prevention interchangeably in this paper. Monitoring level m is a value between 0 and 1 ($0 \leq m \leq 1$).

Hence, an increase in monitoring level m leads to a linearly proportional decrease in incidents. Also, since the total number of faults in the system should remain the same (elaborated in Section 3.5), decrease in incident arrival rate leads to a linearly proportional increase in event arrival rate. At $m = 0$ only incidents arrive and the event arrival rate is zero. Similarly at $m = 1$ only events arrive and the incident arrival rate is considered to be zero.

Cost of Monitoring $C(m)$: Cost of monitoring $C(m)$ is considered to be an increasing function of m. We also consider that $C'(m)$ (Derivative of $C(m)$) is an increasing function of m. This is because increasing the monitoring level m from 0% to 10% would be much cheaper than increasing monitoring level m from 80% to 90%. This cost forms a part of the conformance cost as given in the Juran's COQ model (Juran 1962). The respective substitute of Juran's non-conformance cost is the incident handling cost as it could not be prevented. Also, the cost of monitoring would be 0 when monitoring level m is 0 ($C(0) = 0$) and the cost of monitoring would be ∞ when monitoring level m is 1 ($C(1) = \infty$). Therefore, we choose the $C(m)$ as follows

$$C(m) = \frac{1}{\alpha} ln\left(\frac{1}{1-m}\right) \qquad (1)$$

The Event management is modelled as an M/M/c queuing model. Resources handling the events are viewed as the servers in the queuing model. Events arrive with a Poisson distribution into the queue and wait to be handled by one of the resources.

Event Arrival Rate $\lambda_E(t, m)$: The event arrival process is modelled by Poisson process with rate $\lambda_E(t, m)$. Events can be further divided into true events and false events given by rates $\lambda_{Etrue}(t, m)$ and $\lambda_{Efalse}(t, m)$ respectively. False events can be viewed as false alerts generated by the monitoring sensors. Although, the resolution time for both true and false events is identically distributed, false events do not affect the number of changes made, whereas true events do.

True event arrivals should go down as application maturity increases. Hence rate $\lambda_{Etrue}(t, m)$ should decrease with time t. Also, the arrival of true events should go up as monitoring increases. Hence $\lambda_{Etrue}(t, m)$ increases with m.

We choose the following function for $\lambda_{Etrue}(t, m)$:

$$\lambda_{Etrue}(t, m) = a_2 \times e^{-b_2 \times t} \times m \qquad (2)$$

The false event arrivals remain independent of application maturity as removing faults in application does not necessarily decrease the false notifications, which are due to the improper configuration of the monitoring tool. Hence rate $\lambda_{Efalse}(t, m)$ remains constant with time t. Also the arrival of false events should go up as monitoring increases. Hence $\lambda_{Efalse}(t,m)$ increases with m. The following function is chosen for $\lambda_{Efalse}(t,m)$:

$$\lambda_{Efalse}(t,m) = m \times k \qquad (3)$$

Hence the event arrival rate $\lambda_E(t,m)$ is given as follows:

$$\lambda_E(t,m) = \lambda_{Etrue}(t,m) + \lambda_{Efalse}(t,m) \qquad (4)$$

$$= a_2 \times e^{-b_2 \times t} \times m + m \times k \qquad (5)$$

Splitting Probability P_E: We assume that only a certain fraction of true events arriving at event management go to problem management. Typically, in IT service operations multiple events are generated from the same root cause. Probability P_E is the probability that a true event departing from event management goes to a problem management. According to the Burke's (1956) theorem, the departure process from an M/M/C queue is a Poisson process with parameters same as the parameters for arrival rate. Combining Burke's theorem with Poisson splitting (Ross 2014), we get the arrival rate of events for problem management.

$$\text{the event arrival rate for problem management } = \lambda_{Etrue}(t,m) \times P_E \qquad (6)$$

Event Resources: As mentioned earlier, the resources $R_E(t)$ correspond to the servers in a given time period t. The cost of the server for one staffing-interval is C_E. We model only the minimum (optimum) number of resources ($R_E{}^*(t,m)$) needed at each block (Event Management, Incident Management, Problem Management, Change Management) for a given monitoring level m and time period t. To arrive at the optimal resource configuration, we optimize for every time period using the model presented in Section 3.6.

We assume that resolution time of events is independent and identically distributed across different resources. The resolution time of events by these resources is modelled to be exponentially distributed with rate μ_E. In every staffing interval more than $P_{SLA,E}$ fraction of events should be handled within $t_{SLA,E}$ time.

3.2 Incident Management

An unplanned interruption that can impact an IT service' availability and/or quality is referred to as an incident (Steinberg 2013). An un-intercepted event can become an incident. Disk is 'nearly full' is an 'event' whereas 'disk is full' is an incident. Incident Management is the process for dealing with all incidents (including failures), questions or queries from the users, by technical staff, or automatically detected and reported by event monitoring tools (Cannon et al 2007). The primary goal of the Incident Management process is to 'Quick fix' the incident, restore normal service operation and minimize the impact on business operations. Systems' thinking discourages 'quick fixing' (Braun 2002, Senge 1990). However, in incident management 'Normal service operation' is defined as service operation restored within SLA limits (Cannon et al 2007). Service availability requirements are so stringent that 'quick fixing' may be the only way to meet those norms.

Similar to the event management block, incident management is modelled as an M/M/c queuing model.

Incident Arrival Rate $\lambda_I(t,m)$: The incident arrival process is modelled by Poisson process (Ross 2014) with rate $\lambda_I(t,m)$. The incident arrivals goes down as application maturity increases. Hence, rate $\lambda_I(t,m)$ decreases with time t. Also the arrival of incidents should go down as monitoring increases. Hence $\lambda_I(t,m)$ decreases with m. The following function is chosen for $\lambda_I(t,m)$:

$$\lambda_I(t,m) = a_1 \times e^{-b_1 \times t} \times (1-m) \qquad (7)$$

Splitting Probability P_I: We assume that only a certain fraction of incidents arriving at incident management go to problem management. Typically, in IT service operations multiple incidents are generated because of the same root cause. Probability P_I is the probability that an incident departing from incident management goes to a problem management. Hence similar to the derivation of event arrival rate for problem management, combining Burke's theorem (Burke 1956) and Poisson splitting (Ross 2014), we get the incident arrival rate for problem management.

$$\text{incident arrival rate for problem management} = \lambda_I(t,m) \times P_I \qquad (8)$$

Incident Resources: The number of resources $R_I(t)$ correspond to the number of servers in the time interval t. The cost of the server for one staffing-interval is C_I. Similar to event management block, ${R_I}^*(t,m)$ in our model corresponds to the the optimal number of resources employed at the incident management block for a given monitoring level m and time period *t*.

We assume that resolution time of incidents is independent and identically distributed for all the resources. The resolution time of incidents by these resources is modelled to be exponentially distributed with rate μ_I. Incident management has very stringent SLA requirements. In every staffing interval more than $P_{SLA,I}$ fraction of incident should be handled within $t_{SLA,I}$ time.

3.3 Problem Management

ITIL (Cannon et al. 2007) describes a 'problem' as the unidentified source of one or more incidents/events. Problem Management is the process responsible for managing the lifecycle of all problems. Objectives of Problem Management is to conduct root cause analysis on incoming incidents and events, eliminate the impact of recurring incidents/events and to minimize the impact of incidents that cannot be prevented. Problem Management works together with Event Management, Incident Management and Change Management to ensure service availability and improve service quality. Problem management consists of two sub-processes.

Reactive problem management: Problem management usually implemented as part of Service Operation– and is done as a reaction to any service outage (Cannon et al 2007).

Proactive problem management: Problem management which is initiated in Service Operation, but generally driven as part of Continual Service Improvement. Events that represent situations where the appropriate response will need to be handled by problem management process by creating a problem record for root cause analysis (Case 2007).

Similar to previous blocks, Problem Management is also modelled as an M/M/c queuing model.

Problem Arrival Rate: Problem arrival rate is given as the sum of rate of incidents arriving for problem management and the sum of the rates of events arriving for problem management (see Equations (8) and (6)). Hence problem arrival rate is given by,

$$\begin{aligned}\text{Problem arrival rate} &= \lambda_I(t,m) \times P_I + \lambda_{Etrue}(t,m) \times P_E \\ &= a_1 \times e^{-b_1 \times t} \times (1-m) \times P_I + a_2 \times e^{-b_2 \times t} \times m \times P_E\end{aligned}$$

Splitting Probability P_P: We assume that only a certain fraction of problems arriving at problem management go to change management. Often in IT service operations only a fraction of the problems are approved for changes. The probability P_P is the probability that a problem departing from problem management goes to a change management. Similar to previous sections, combining Burke's theorem (see Burke 1956) and Poisson splitting (see Ross 2014) the arrival rate for change management is

$$\text{Arrival rate for changes} = (\lambda_I(t,m) \times P_I + \lambda_{Etrue}(t,m) \times P_E) \times P_P. \quad (9)$$

Problem Resources: The number of resources $R_P(t)$ correspond to the number of servers in the time interval$t$. The cost of the server for one staffing-interval is C_P. Similar to other blocks, ${R_P}^*(t,m)$ in our model corresponds to the optimal number of resources employed at the problem management block for a given monitoring level m and time period *t.* We assume that resolution time of problems is independent and identically distributed across different resources. The resolution time of problems by these resources is modelled to be exponentially distributed with rate μ_P. In every staffing interval more than $P_{SLA,P}$ fraction of problems should be handled within $t_{SLA,P}$ time.

3.4 Change Management

The purpose of change management is to respond to the customer's changing business requirements while maximizing value and reducing incidents, disruption and rework. To achieve this it uses standardized approaches and procedures for handling the changes in order to minimize the number and impact of any related incidents upon service. All changes to service assets and configuration items are recorded in the Configuration Management System (Cannon et al 2007).

Similar to other blocks, the change management is also modelled as an M/M/c queuing model.

Change Arrival Rate: Change arrival rate is given by Equation (9). All the problems arriving in change management lead to changes which increases the application maturity.

Change Resources: The number of resources $R_C(t)$ correspond to the number of servers in the time interval t. The cost of the server for one staffing-interval is C_C. Similar to other blocks, $R_C{}^*(t,m)$ in our model corresponds to the the optimal number of resources employed at the change management block for a given monitoring level m and time period *t.*

We assume that resolution time of changes is independent and identically distributed across different resources. The resolution time of changes by these resources is modelled to be exponentially distributed with rate μ_C. In every staffing interval more than $P_{SLA,C}$ fraction of problems should be handled within $t_{SLA,C}$ time.

3.5 Application Maturity $\boldsymbol{Z(t,m)}$

Intuitively, application maturity can be thought of as the measure of how bug free an application is. Mathematically, application maturity $Z(t,m)$ is defined as the expected time between faults, which are incidents and true events. Initial application maturity is defined as the application maturity at the start of the service. t_{start} is defined as the number of months the application has matured prior to the current service. The variable assumes applications start at zero maturity and matures over t_{start} at the same rate.

Application maturity ($Z(t,m)$) is inversely proportional to the error proneness of the application ($E(t,m)$), ($Z(t,m) = E(t,m)^{-1}$). The error proneness of an application $E(t,m)$ can be described as the rate at which the faults occur (incidents and true events).

$$E(t,m) = a_1 \times e^{-b_1 \times t} \times (1-m) + a_2 \times e^{-b_2 \times t} \times m. \quad (10)$$

Such that following equality holds,

$$\frac{a_1}{1-e^{-b_1 \times t}} \times P_I = \frac{a_2}{1-e^{-b_2 \times t}} \times P_E. \quad (11)$$

Also $a_1 \times e^{-b_1 \times t} \times (1-m) \times P_I \times P_P \times P_C$ describe the expected number of changes made by incidents in time interval t. Similarly $a_2 \times e^{-b_2 \times t} \times m \times P_E \times P_P \times P_C$ describe the expected number of changes made by events in time in interval t. The total number of expected changes over a large horizon should remain same for all. Hence we want the total number of changes made when $m = 0$ to be equal to the changes made when $m = 1$ for $T = \infty$. Thus we have,

$$\sum_{1}^{\infty} a_1 \times e^{-b_1 \times t} \times P_I \times P_P \times P_C = \sum_{1}^{\infty} a_2 \times e^{-b_2 \times t} \times P_E \times P_P \times P_C.$$

Hence we require equality given in equation 11. This model of application maturity is similar to software reliability model used in Goel et al 1979.

Application criticality is another critical variable for IT production support service engagements. Although an explicit variable to capture application criticality has not been considered in this model, SLA requirements can adequately proxy for application criticality. For example, a critical application could be the application to facilitate high frequency trading in an investment bank. Such applications tend to have stringent SLAs as compared to an HR or a Payroll application.

3.6 Optimal Resource Computation $\boldsymbol{R_X}^*(\boldsymbol{t},\boldsymbol{m})$

We assume that the manager always employs optimal number of resources needed to honour SLA requirements. To capture this assumption in our model, the optimal number of resources at each block- incident, event, problem and change management are computed by the formulation below.

$$R_X{}^*(t,m) = argmin_{R_X(t)} C_X \times R_X(t) \quad \forall X \in \{I,E,P,C\} \qquad (12)$$

$$\text{s.t.} \qquad R_X(t) \in Z^+$$

$$W_X(t_{SLA,X}, R_X(t), t, m) \geq P_{SLA,X}$$

In this optimization model the objective function is to minimize the resource costs and the constraint is to satisfy SLA norms. We used the algorithm proposed by Kontogiorgis & Tibbs (2005) for solving the optimization problem 12. Here $W_x(t_{SLA,x}, R_x(t), t, m)$ is probability that the waiting time in an M/M/R_X queuing model does not exceed $t_{SLA,I}$ and is given as follows (Gross & Harris 1998):

$$W_x(t_{SLA,x}, R_x(t), t, m) = \frac{r(t,m)^{R_x(t)}\left(1 - e^{-\mu_x\left(R_x(t) - r(t,m)\right)t_{SLA,x}}\right)}{(R_x(t) - 1)!\left(R_x(t) - r(t,m)\right)} p_0(R_x(t)) + W_x(0, R_x(t)$$

where $r(t,m) = \lambda_x(t,m)/\mu_x$ is the offered load,

$$W_x(0, R_x(t)) = 1 - \frac{R_x(t) r(t,m)^{R_x(t)}}{R_x(t)!\,(R_x(t) - r(t,m))} p_0(R_x(t))$$

$$p_0(R_x(t)) = [\sum_{n=0}^{R_x(t)-1} \frac{r(t,m)^n}{n!} + \frac{r(t,m)^{R_x(t)}}{R_x(t)!} \frac{R_x(t)}{R_x(t) - r(t,m)}]^{-1}.$$

3.7 Computation of optimal monitoring level $\boldsymbol{m}^*$

Instead of using a continuous set for m between 0 and1, we work with a discrete set $\mathcal{M}$ between 0 and 1. This makes the optimization problem much simpler. This approach is justified because exact value of m is not very useful for practical purposes. Also we can be fairly close to the optimal solution by considering a fairly detailed set of m.

For all these values of $m \in \mathcal{M}$ and $\forall t \leq T$, our model will only allocate the optimal number of resources $R_X{}^*(t,m)$ (see Equation 12) at incident management, problem management, event management and change management.

We therefore get

$$C_X{}^*(m) = \sum_{t=1}^{t=T} C_X \times R_X{}^*(t,m) \quad \forall X \in \{I,E,P,C\}. \qquad (14)$$

We consider

$$C_R{}^*(m) = \sum\nolimits_{X \in \{I,E,P,C\}} C_X^*(m)$$

We define total cost

$$C_{Total}(m) = C_R{}^*(m) + C(m) \quad (15)$$

Hence the optimal monitoring level is given as,

$$m^* = argmin_{m \in \mathcal{M}} C_{Total}(m) \quad (16)$$

4 RESULTS AND DISCUSSION

To illustrate the formulation, we have used relevant data from an actual ITSM engagement for an American market research firm, handled by one of the largest IT service providing firm in the world. This study has been conducted in the context of an IT operations support engagement outsourced by a client to a service provider in managed services mode. In managed services mode of outsourcing the responsibility of managing the engagement and carrying out project management activities within the engagement lies with the service provider. The person managing the engagement has the prerogative of deciding on resource and cost management policies in order to meet compliance requisites and larger engagement goals. Data is used to calibrate the functions in our mathematical model. Following are the input parameters to our model.

Time Horizon: We look at a service which spans over $T = 24$ months. Each month consists of 30 days. Each day has one 8 hour shift. We take one month as a staffing interval. Also, the changes to application are deployed as patches at the end of every month. Consequently, the incident and event rate change after every month. One month is taken as one unit of time.

Event Management:

Cost of monitoring $C(m)$: Based on the data from service engagement, we have calibrated the function below to cost $\$ .15$ million at 90% monitoring level. The figure 90% is based on a broad assumption. To relax this assumption we have done sensitivity analysis to see the impact of parameter α on optimal monitoring level m and the cost at optimal monitoring level (see Section 4.2).

$$C(m) = 65144.17 \times log(1/(1 - \mathrm{m}))\$$$

Arrival rates: True event arrival rate is given as follows

$$\lambda_{Etrue}(t, m) = 5 \times 8 \times 30 \times e^{-0.118 \times t} \times m \text{ true events/ month}$$

Here we assume, a drop rate of around 11% in event arrival rate after every time interval (1 month). To relax this assumption we have done sensitivity analysis to see the effect of drop rates on the optimal monitoring level m (see Section 4.2), false event arrival rate is given as follows

$$\lambda_{Efalse}(t, m) = m \times 10 \times 30 \text{ false events/month}$$

$$\therefore \lambda_E(t, m) = 1200 \times e^{-0.118 \times t} \times m + m \times 300 \text{ events /month.}$$

Splitting probability: $P_E = 0.04$

Resources: Resource cost $C_E = 10 \times 8 \times 30/(\text{event resource} \times \text{ month})$. Resolution rate $\mu_E = 4 \times 8 \times 30$ events/month.

Incident Management:

Arrival rates: $\lambda_I(t, m) = (4/3) \times 300 \times e^{-0.15 \times t} \times (1 - m)$ incidents /month

Here we assume a drop rate of around 14% in incident arrival rate after every time interval. To relax this assumption we have done sensitivity analysis to see the effect of drop rates on the optimal monitoring level m (see Section 4.2).

Splitting probability: $P_I = 0.15$

Resources: Resource cost $C_I = 25 \times 8 \times 30/(\text{Incident resource} \times \text{month})$

Resolution rate $\mu_I = 4 \times 30$ incidents /month

Problem Management:

Splitting probability: $P_P = 0.8$

Resources**:** Resource cost $C_P = 50 \times 8 \times 30/(\text{Problem resource} \times \text{month})$

Resolution rate $\mu_p = 1 \times 30$ problems /month

Change Management:

Resources**:** Resource cost $C_C = 80 \times 8 \times 30/(\text{Change resource} \times \text{month})$

Resolution rate $\mu_c = 10$ changes /month

SLA Requirements: SLA requirements are as follows:

X	$P_{SLA,X}$	$t_{SLA,X}$
Event Management	0.95	$2 \times (\mu_E)^{-1}$
Incident Management	0.95	$1.1 \times (\mu_I)^{-1}$
Problem Management	0.95	$2 \times (\mu_P)^{-1}$
Change Management	0.95	$2 \times (\mu_C)^{-1}$

Application Maturity *Z(t,m)*: $E(t,m) = 400 \times e^{-0.15 \times t} \times (1-m) + 1200 \times e^{-0.118 \times t} \times m$. Note that the equality in equation 11 is satisfied. We compute the optimal monitoring level by following the methodology in Section 3.7, we take values of m in the set $\mathcal{M} = \{.0000, .0001 \ldots, .9999\}$.

4.1 Total Cost vs Monitoring Level

In Figure 2 the total cost ($C_{Total}(m)$) is plotted against the monitoring level m. Broadly, $C_{Total}(m)$ versus Monitoring level follows a ’U’ shaped curve.

Analogy with Juran’s Model: The faults described in Juran’s (1962) model correspond to incidents here. Increased quality in Juran’s model or increased prevention level here results in increased conformance. Hence, the level of prevention is analogous to quality defined in Juran’s model. The ’U’ shaped pattern between total cost and monitoring level is similar to the relation betwen cost of quality and quality in Juran’s model, this broadly validates our results.

Also, the variation in total cost with respect to monitoring level is discrete and not smooth. The discreteness can be explained by a deeper analysis of different cost components. Total cost is the sum of resource costs (incident, problem, event, change management resources) over the entire service engagement ($C_R{}^*(m)$) (see Figure 3) and the onetime cost of monitoring $C(m)$.

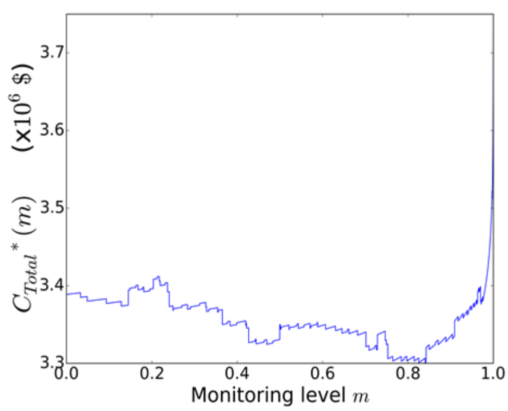

Figure 2: Total Cost vs Monitoring level

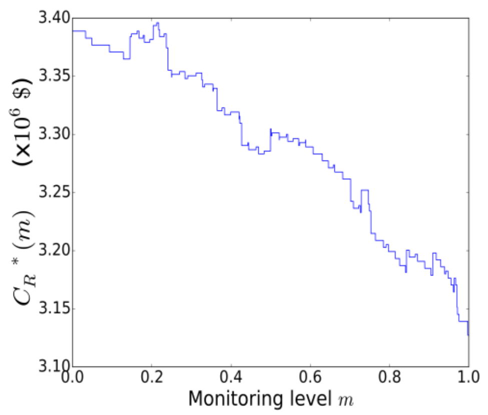

Figure 3: Total Resource cost (For optimal resources) vs Monitoring level

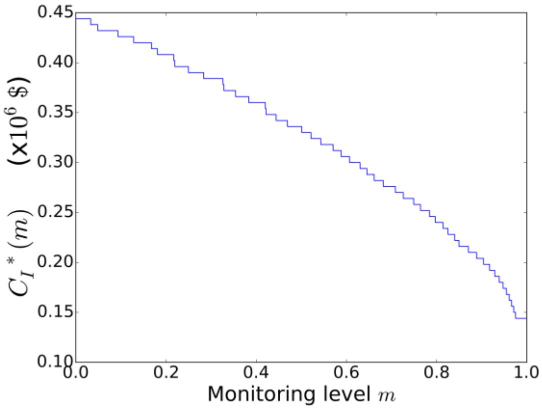

Figure 4:Optimal Incident Mangement Cost vs Monitoring level

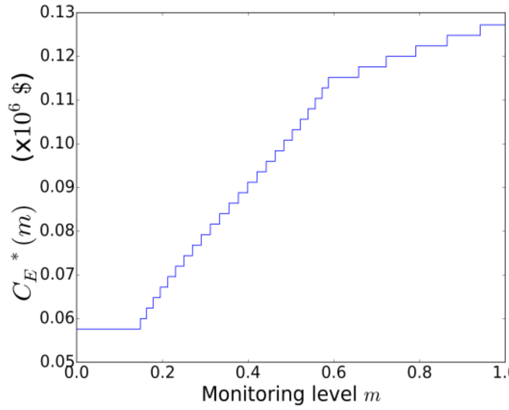

Figure 5:Optimal Event Mangement Cost vs Monitoring level

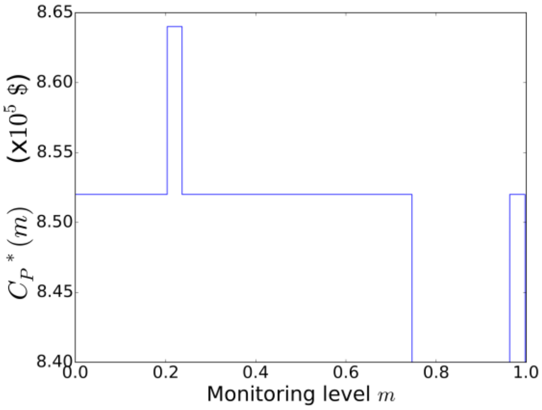

Figure 6:Optimal Problem Mangement Cost vs Monitoring level

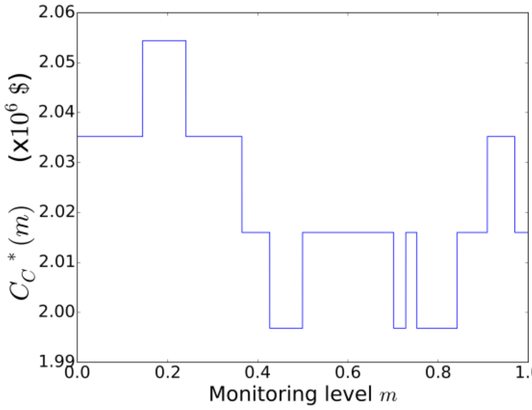

Figure 7:Optimal Change Mangement Cost vs Monitoring level

Figures 4, 5 show the Incident management cost $C_I^*(m)$ and Event management cost $C_E^*(m)$. The abrupt changes in cost of event management and incident management can be explained by a phenomenon called staffing stickiness.

Staffing Stickiness, in this study, refers to the situation where the number of resources cannot be adjusted as per the incident/event arrival rates on a real time basis. For example, due to deployment of a patch (changes) onto the application, the incident/event arrival rates may have reduced. But, unless the quantum of work reduced is large enough for one or more resources to be released while satisfying the SLA requirements, the number of resources remain the same. In addition to the SLA requirements, administrative delays involved in on-boarding resources (hiring, training etc.) could cause delay in maintaining optimal number of resources. This is similar to the concept of nominal rigidity, often referred to as price/wage stickiness, in economics (Rankin 1998).

Figures 6, 7 denote the Problem management cost $C_P^*(m)$, Change management cost $C_C^*(m)$. We can see that there are non-monotonic and abrupt changes in both these figures. The non montonic nature exists because the resource costs at problem management and change management increase with incident and event arrival rates, but incident arrival rate decreases with *m* and event arrival rate increases with *m*, in addition, the rates at which these decrease and increase are different, thus creating non-monotonic changes in both problem management and change management costs. The abruptness can again be explained by the staffing stickiness.

A more detailed explanation for the non-monotonic nature of these graphs is as follows. Since the cost of both Problem management and Change management depend on the resources deployed (${R_P}^*(t,m)$ and ${R_C}^*(t,m)$ respectively) at these blocks. These resources deployed are increasing step functions of arrival rates at the problem management and change management block. The arrival rates at both these blocks are directly proportional to the sum of the departure rates from incident and event management blocks. The departure rates of both incident management and event management block follow an exponentially decaying pattern with respect to time. Because of equality in equation 11 the function of departure rate from incident management at $m = 0$ with respect to time t will always intersect the function of departure rate from event management at $m = 1$ with respect to time t at some time $\tilde{t}$. Because of which the arrival rate at the problem and change management block will always decrease (increase) with m before the time $\tilde{t}$ after which it will increase (decrease) with m. Hence the resources ${R_P}^*(t,m)$ and ${R_C}^*(t,m)$ decrease (increase) with m before $\tilde{t}$ and increase (decrease) with m after $\tilde{t}$. As long as this $\tilde{t}$ is less than the time horizon T this pattern can be observed.

4.2 Sensitivity Analysis

We conduct sensitivity analysis on some parameters of the model to see their impact on the optimal monitoring level. The choice of these parameters is due to the fact that these parameters could not be objectively derived from an engagement. The monitoring cost is directly proportional to $1/\alpha$ resulting in a cheaper monitoring system at higher α. Hence optimal monitoring level would always increase with α (Figure 8). However, the interesting aspect in Figure 8 is the presence of plateaus in the graph. These plateaus are due to changes in α not impacting resource costs or arrival rates. This means that the resource cost function (${C_R}^*(m)$) would remain exactly the same (as in Figure 3). As observed in Figure 3, ${C_R}^*(m)$ has some local troughs between some ranges of m, when the optimal monitoring level m^* is between these ranges these plateaus are observed in final graph (Figure 8). From Figure 8, we can see that the cost at the optimal monitoring level (refer to the dotted line) decreases as α increases. Also, for initial values of α the cost is constant since the monitoring level is 0. The phases in the graph in which the monitoring rate is constant, the cost decreases such that slope at which it is decreasing also decreases with respect to α. This is because the cost is directly proportional to $1/\alpha$ in that case.

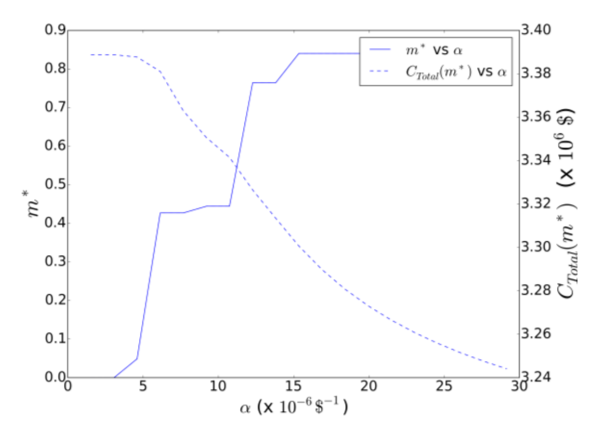

Figure 8: Optimal monitoring level and Cost at optimal monitoring level vs α

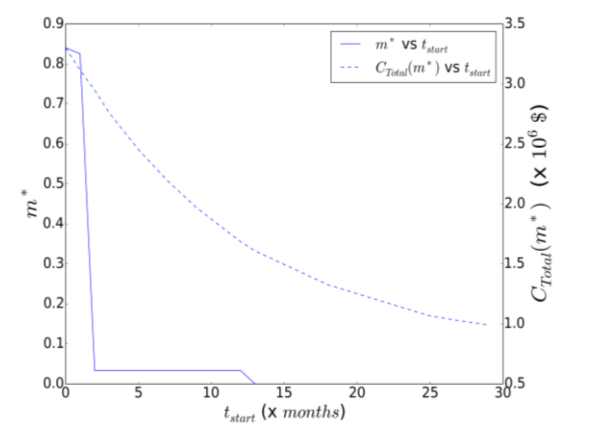

Figure 9: Optimal monitoring level and Cost at optimal monitoring level vs t_{start}

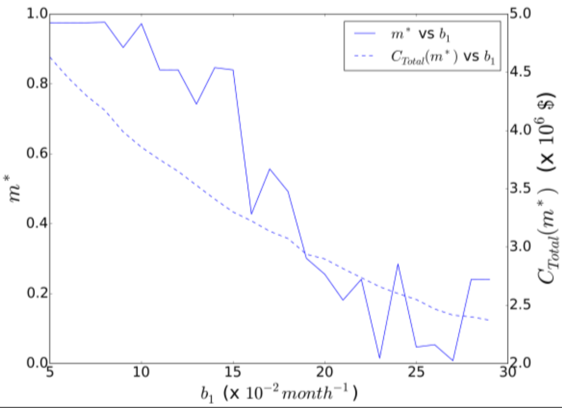

Figure 10: Optimal monitoring level and the cost at optimal monitoring level vs b_1

The graphs in figure 9 correspond to sensitivity analysis of Optimal monitoring level with Initial application maturity. An application that has been previously supported is expected to have gone through changes and thus, more stable (higher t_{start}). Whereas, a greenfield support project, just out of development, is generally more error prone (lesser t_{start}). From Figure 9 we can see that the optimal monitoring level declines in an abrupt manner with t_{start}. To explain the abruptness, we refer to Figure 2, where the costs at all the monitoring levels are fairly close to each other before they take off at monitoring levels close to 1, hence little changes in the Initial application maturity changes m^* dramatically.

We vary the drop rate b_1 to observe changes in m^*. b_1 essentially describes how quickly incident rates decrease as application matures. All other parameters are kept constant except b_2, which signifies the drop in event arrival rate, to satisfy the equality presented in equation 11. From Figure 10 we can see that the optimal monitoring level follows a zig-zag pattern with a general trend of decline in optimal monitoring level as b_1 increases. This is because as b_1 increases b_2 also increases. Increase in b_1 leads to faster decrease in incident arrival rates for any m (see equation 7), which leads to lesser amount of resources required for incident management, but increase in b_2 leads to lesser amount of resources required for event management this explains the zig-zag nature of Figure 10. Also, since the incident resource cost is relatively higher than the event resource cost the general trend is for the monitoring level to go down.

5 CONCLUSION

An all-out preventive approach (m = 1) has the potential to ensure 100% service availability. Having said that the abnormally high cost of deploying a monitoring system that intercepts every possible threat to service availability, confines us to live with downtimes of acceptable proportions (Burrin et al 2007). In this study, we attempted to model the trade-off between the cost of conducting service operations and the level of prevention. The model was then tuned to parameters from an actual engagement. We observed a ’U’ shaped curve characterized by small discrete steps between cost and prevention level. Whereas the shape of the curve confirms the analogy between this study and Juran’s cost of quality model (Juran 1962), the discreteness is due to a phenomenon called Staffing Stickiness, which is analogous to the concept of nominal rigidity (Rankin 1998). Although, the model’s input parameters were mostly derived from an actual engagement, there are certain parameters that cannot be objectively quantified and hence could not be taken directly from the engagement. To overcome this limitation, we conducted sensitivity analysis on these variables to assess their impact on the optimal monitoring level and cost of operations. An interesting observation from the sensitivity analysis results is the presence of plateaus, where optimal monitoring level remains constant despite changes in monitoring cost. For practical applicability of this study, an important prerequisite is to define an objective measure for monitoring level. Although, such a measure is beyond the boundaries of this study, it could be an interesting direction to extend this research.

References

Burke, P. J. (1956). The output of a queuing system. Operations research, 4(6), 699-704.
Ross, Sheldon M. Introduction to probability models. Academic press, 2014.
Goel, Amrit L., and Kazu Okumoto. “Time-dependent error-detection rate model for software reliability and other performance measures.” IEEE transactions on Reliability 3 (1979): 206-211.
Kontogiorgis, S., and Tibbs, R. W. (2005, March). An efficient approximation to wait time in M/M/c queues with application to staffing planning. In Proceedings of the 43rd annual Southeast regional conference-Volume 2 (pp. 98-102). ACM.
Gross, D., and Harris, C. (1998). Fundamentals of queueing theory.
Juran, J. M. (1962). The economics of quality. Quality Control Handbook, New York, 1-31.
Rankin, N. (1998). Nominal rigidity and monetary uncertainty. European Economic Review, 42(1), 185-199.
William Braun (2002). The System Archetypes. Copyright © 2002 by William Braun (2002.02.27).
Peter Senge (1990). The Fifth Discipline (New York: Doubleday Currency), 1990, 142, 150-55, 156, 226.
Cannon, D., Wheeldon, D., Taylor, S., & Office of Government Commerce Great Britain. (2007). ITIL:[IT service management practices; ITIL v3 core publications].[4]. Service operation. TSO (The Stationery Office).
Case, G. (2007). ITIL v3. Continual Service Improvement. Office of Government Commerce.
Steinberg, R.A. (2013). Measuring ITSM: Measuring, Reporting, and Modelling the IT Service Management Metrics that Matter Most to IT Senior Executives. USA: Trafford Publishing
Van Bon, J., Pondman, D., and Kemmerling, G. (2002). IT Service Management: An Introduction. Amstredam: Van Haren Publishing
Papazoglou, M. P. (2008, January). The challenges of service evolution. In Advanced Information Systems Engineering (pp. 1-15). Springer Berlin Heidelberg.
Burrin, N., Regev, G., and Wegmann, A. (2007, June). Adapt and Adopt: An Experiment in Making Best Practices Adequate in an Organization. In 19th International Conference on Advanced Information Systems Engineering, Trondheim, Norway.
Shavell, S. (1984). A model of the optimal use of liability and safety regulation. The RAND Journal of Economics, 15(2), 271-280.
Kolstad, C. D., Ulen, T. S., & Johnson, G. V. (1990). Ex post liability for harm vs. ex ante safety regulation: substitutes or complements?. The American Economic Review, 888-901.
Wittman, D. (1977). Prior regulation versus post liability: The choice between input and output monitoring. The Journal of Legal Studies, 6(1), 193-211.
Kleindorfer, P. R., & Saad, G. H. (2005). Managing disruption risks in supply chains. Production and operations management, 14(1), 53-68.
Hendricks, K. B., & Singhal, V. R. (2005). An empirical analysis of the effect of supply chain disruptions on long-run stock price performance and equity risk of the firm. Production and Green, L. V., Kolesar, P. J., & Whitt, W. (2007). Coping with Time-Varying Demand When Setting Staffing Requirements for a Service System. Production and Operations Management, 16(1), 13-39.Operations management, 14(1), 35-52.
Chopra, S., & Sodhi, M. S. (2004). Managing risk to avoid supply-chain breakdown. MIT Sloan management review, 46(1), 53.
Whitt, W. (1999). Dynamic staffing in a telephone call center aiming to immediately answer all calls. Operations Research Letters, 24(5), 205-212.